Network Inference by Combining Biologically Motivated Regulatory Constraints with Penalized Regression

Fabio Parisi,[a,b] **Heinz Koeppl,**[c] **and Felix Naef**[a,b]

[a]*School of Life Science, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland*

[b]*Swiss Instiutute of Bioinformatics, Lausanne, Switzerland*

[c]*School of Computer and Communication Sciences, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland*

Reconstructing biomolecular networks from time series mRNA or protein abundance measurements is a central challenge in computational systems biology. The regulatory processes behind cellular responses are coupled and nonlinear, leading to rich dynamical behavior. One class of reconstruction algorithms uses regression and penalized regression to impose sparseness on the solution, as requested biologically. Motivated by the five-gene challenge in the Dialogue for Reverse Engineering Assessments and Methods 2 (DREAM2) contest, we extend and test penalized regression schemes both on data from simulations and real qPCR measurements. The methods showing best performance are the Adaptive Ridge (AR) regression and a new extension thereof, in which we impose a biological constraint to the reconstructed network. Specifically, we request from the solutions that the outgoing links have the same regulatory sign, which is a reasonable approximation for most prokaryotic transcriptional networks. In other words, a given regulator must be either an activator or a repressor but not both. The constraints can be implemented with quadratic programming, and we show that this improves the reconstruction performance significantly. While linear models are not sufficiently general to encompass most complex behaviors, they offer powerful tools for network reconstruction, particularly for systems operating near a steady state. In particular, the optimization problems are well behaved and methodologies allow finding global optima efficiently. Adding constraints reflecting biological circuit designs is one of the most important aspects of network inference. We propose one such constraint, namely the consistency in the signs of outgoing links, which will facilitate the inference of transcriptional regulatory networks.

Key words: **network inference; regulatory networks; penalized regression**

Introduction

The recent acquired ability to perform large-scale quantitative expression or activity measurements in well-controlled biological systems has opened the possibility that a system's logic may be recovered using reverse engineering principles. An ambitious goal is to learn causal relationships between proteins or genes from systematic time-series data. For example, we would like to learn from kinetic data which are the direct targets of a transcription factor or a protein kinase. While this seems difficult presently for comprehensive whole-genome networks, the present manuscript explores the possibility that it can be successful for smaller networks, typically less than 50 genes.

Address for correspondence: Felix Naef, School of Life Sciences, Station 15, Ecole Polytechnique Federale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland. Voice: +41-21-693-1621. felix.naef@epfl.ch

The Challenges of Systems Biology: Ann. N.Y. Acad. Sci. 1158: 114–124 (2009).
doi: 10.1111/j.1749-6632.2008.03751.x © 2009 New York Academy of Sciences.

One class of experimental data used for the reconstruction of genetic networks consists of perturbation experiments through gene knockout and gene suppression, or through environmental stress.[1] A very broad spectrum of models and algorithms has been applied to the gene-network inference problem. Some of them originate from the classical description of biochemical reaction kinetics, such as *mass action kinetics*, *Michaelis–Menten type kinetics*, or the *power law approximation*.[2] Other models abstract from the biochemical details and deal with dependencies, applying cluster analysis[3] or using purely graph-theoretical tools,[4] or combine the latter with conditional probability distributions to describe these dependency structures.[5]

A statistical approach that was designed to uncover the biochemical details of a network is the *correlation metric construction*[6] and its generalization in terms of mutual information.[7] In the same flavor as the correlation metric construction but abstracted from the biochemical details are *relevance networks* and their related algorithm *ARACNE* (Algorithm for the Reconstruction of Accurate Cellular Networks).[8] Both methods estimate the pair-wise mutual information between gene expression levels, whereas the latter additionally applies the data processing inequality to distinguish between direct and indirect interactions. In general, methods based on a correlation analysis share the property that they cannot infer the causality structure of the network, that is, they are only able to infer a structure in terms of an undirected graph. Other variants that infer an undirected graph are Gaussian graphical models. Graphical models are state-of-the-art models for data analysis in computer science and have been successfully applied to the inference of regulatory networks.[9] In the case of Gaussian graphical models, it boils down to estimating the covariance matrix and partial correlation coefficients of the gene expression levels. In contrast to the previous statistical methods, *Bayesian networks*[9] are able to infer causality (or at least directionality), but are restricted to the class of acyclic directed graphs. Cycles such as regulatory feedback loops have to be unrolled in time, reminiscent of a technique applied to recursive neural networks. Such unrolled models are called *dynamic Bayesian networks* and are widely applied nowadays (see Ref. 5 for a comparative study of different inference algorithms). A special member of the class of dynamic Bayesian networks is the Kalman filter. It has also been applied to the inference of gene regulatory networks.[10] Another model class for gene regulatory networks includes deterministic Boolean and finite state networks[11] and their probabilistic generalizations.[12] They are based on the observation that synthetic gene networks have been shown to implement different kinds of Boolean functions.

Motivated by biochemical reaction kinetics, methods for reconstruction based on ordinary differential equations have been proposed.[13] In particular, linear differential equations or *additive regulation models* are frequently used.[14,15] The assumption for these linear models is that the applied perturbation to the regulatory network is small such that the underlying nonlinear differential equation can be linearized around an operating point.[16] The estimated Jacobian matrix reflects the dependency structure and can be associated with the adjacency matrix of the graph of the biochemical network. Biochemical data poses two main limitations to reconstruction by ordinary least squares (OLS) regression: they contain noise on both the control and the measured variables and they generally correspond to a sparse adjacency matrix. Total least squares (TLS) and penalized regressions were applied to address the problem of noisy data matrices[17] and inferred networks that are overly connected and thus biologically unrealistic.[18,19] Reference 20 demonstrated the equivalence of TLS and ridge regression (RR). Further extension of RR is the adaptive ridge regression (AR). AR balances the penalization on each parameter in the model; it was shown to produce estimates equivalent to the naturally sparse L1-penalized regressions,[21,22] and a convenient expectation maximization implementation has been proposed to solve AR using a hyperparameter to tune the global model complexity.[22]

Interestingly, the majority (>60%) of transcription factors listed in RegulonDB[23] are either activators or repressors, and less than 20% of regulators are activators for some genes and repressors for others. The remaining factors are reported as activating or repressing depending on the conditions. However, 68% of these were inferred by sequence homology with humans. Supported by these considerations, we introduce biologically relevant constraints on the signs of regulatory interactions. This is a key property of our method that significantly improves reconstruction accuracy.

Here, we extend and compare previous penalized regression algorithms (RR and AR) for the reconstruction of genetic networks from time-series data. We focus on linear models; these are suited when the network operates near steady state, providing a computationally fast and powerful framework to dissect dynamical properties of biological systems. We start by defining ad hoc metrics for the accuracy of reconstructed networks models, then we introduce several levels of penalized regression models (AR with sign constraints) applicable to times-series data and evaluate these algorithms both on artificial and biological network models.

Results

Performance of Reconstruction Algorithms for Randomly Generated Linear Models

We tested the performance of the reconstruction algorithms using randomly generated linear systems with five genes and seven nonzero off-diagonal elements (see Material and Methods, below). The connectivity matrix was further constrained to reflect transcriptional regulation in lower organisms, namely that a transcription regulator is either an activator or a repressor but not both. While there might be exceptions to this rule, the structural constraints imposed are expected to improve model predictions in most cases. In higher organisms, or in the case of post-translational control, this rule may be violated more frequently.

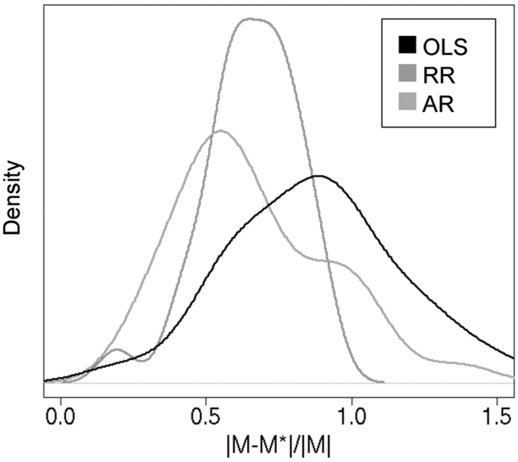

FIGURE 1. Distributions of the relative Frobenius norms for OLS (black), RR (red), and AR (green) for a benchmark over 30 randomly generated five-node networks. OLS performs the worst, and peaked around one, leaving almost all the variance of the matrix M unexplained. RR achieves better results than OLS, with a tight peak around 0.75. On average AR performs the best. The distribution relative to AR, which peaked at 0.5, shows a large right shoulder due to suboptimally inferred networks.

We measured performance using the Frobenius norm of the error in the connectivity matrix estimate (see Material and Methods). As expected, OLS best fits the trajectories, but it is unable to predict sparse connectivity matrices. As result we observed large relative Frobenius norms, often above 100% (Fig. 1). This is due to prediction of links that are absent from the starting matrix and reflects overfitting of the data. RR behaves similarly to OLS, though the penalization term helps reduce the matrix error so that it never exceeds 100% (Fig. 1). As we move toward sparser regression schemes, AR performs better than OLS and RR. The tail with high residual variances is in part due to violation of the sign constraint leading to suboptimal solutions (Fig. 1). On average, for each random M, one link inferred by AR does

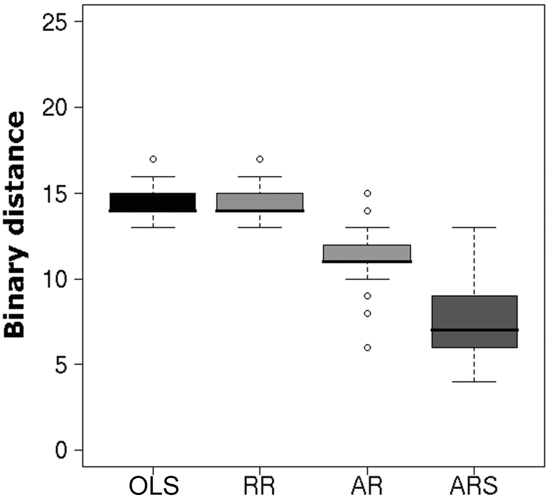

FIGURE 2. Boxplots of the distribution of the absolute binary norms of the inferred networks from the 30 randomly generated five-node networks for the four algorithms (OLS in black, RR in red, AR in green, and ARS in blue). OLS and RR produce comparable results in this metric, mainly due to the zero elements of the real matrix M, corresponding to nonzero elements in the inferred matrix. AR performs better than the previous two, though still containing several links in excess and, on average, one exception to the sign constraint every inferred matrix. ARS outperforms the previous methods. The errors in the ARS-inferred matrix are mainly due to weak links in excess.

not respect the sign rule. This motivated the implementation of a sign-constrained adaptive ridge regression (ARS). When tested against OLS, RR, and AR, ARS shows increased performance most notably in the binary norm (see Material and Methods), which is most sensitive to the correct network topology. As expected, OLS and RR perform poorly in the binary norm, mainly because no links are pruned. AR obtains better results on average, though the violation of the sign rule often causes strong penalizations in the binary norm. ARS outperforms the other three methods, correctly inferring on average 75% of the elements of the matrix M (Fig. 2). The remaining 25% of errors in the ARS are due to weak links with small moduli, which do not contribute much to the Frobenius norm (not shown).

Performance of Reconstruction Algorithms for Two Nonlinear Biological Networks

To test the reconstruction methods in a more generic and biologically relevant setting, we implemented two models taken from the literature. These models have been previously used to asses reconstruction performances of other methods and, in the case of the repressilator, the parameter's influence on stability was reported allowing us to choose a meaningful range of parameter values.

Four-Gene Network

We simulated trajectories in a four-gene network used for benchmark[16] for the equivalent of a 6 h time-course with samplings at regular intervals every 20 min. We tested the accuracy of the reconstructions for increasing noise variance k. We assessed the goodness of the reconstruction by using the binary norm (see Material and Methods). The ARS method has not been applied in this case since node 2 does not respect the sign constraint (Fig. 3A). For small noise levels ($k = 0.1\%$) AR infers correctly 75% of the links in the Jacobian. The performance of AR, however, is affected by the size of the noise. For $k > 10\%$ AR performs equally to OLS or RR in terms of the ε-metric (Fig. 3A). OLS and RR have a steady performance recovering correctly 50% of the elements in the inferred matrix. Moreover, OLS and RR are always better than AR at fitting the trajectories, though the difference decreases with increasing noise levels (not shown). One reason the AR might not perform optimally is that this small network is not very sparse (connectivity is 10 links out of 16 possible).

The Repressilator Network

This three-gene oscillator[24] poses a much greater reconstruction problem, and we do not *a priori* expect optimal reconstruction performance. Namely, in the oscillator regime, the limit-cycle dynamics is truly nonlinear and

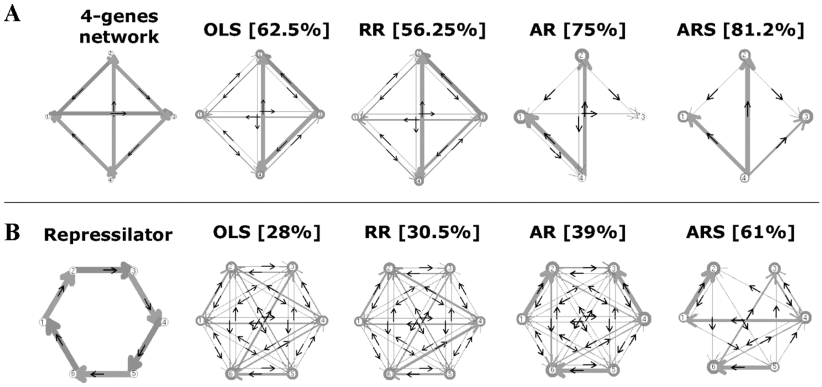

FIGURE 3. Reconstruction performances of *in silico* data of two dynamical systems. Within brackets the percentage of correctly inferred elements according to the binary norm. With the exception of the real network, the thickness of the arrows is proportional to the relative modulus of the corresponding elements within the Jacobian; green indicates induction, and red repression. (**A**) Four-gene network. Real and inferred networks from trajectories. As expected, the results of OLS and RR are affected by overconnectivity. AR, in contrast, captures most of the interactions in the network. (**B**) Repressilator. Nodes 1, 3, and 5 represent the mRNA of the three genes; nodes 2, 4, and 6 represent the regulators encoded by the mRNAs. The nonlinear dynamics of the system causes poor performances in the case of OLS, RR, and AR. ARS, thanks to the sign constraints, correctly emphasizes mRNA translation. Repression by saturation of promoters seems not to be identified.

thus our model does not provide an accurate approximation. Even in the damped regime analyzed below, the repressor functions have hill coefficient of 2, hardly approximated by linear functions. Moreover, promoter saturation is not captured by linear functions. It is nevertheless interesting to see how the proposed reconstruction procedure performs outside its strict range of applicability.

We simulated the trajectories for mRNA and protein levels in the repressilator model (see Material and Methods) and added noise as in the previous cases. AR, OLS, and RR did not perform well, inferring overconnected networks with less than 40% of the correct elements (Fig. 3B). ARS, in contrast, suggested a sparse solution and pointed out the translation of protein from the respective mRNA as the strongest links in the inferred network (Fig. 3B). As speculated, saturation effects, such as promoter saturation by the repressor reducing the mRNA expression, were not properly captured.

DREAM2 Five-Gene Contest

As part of the Dialogue for Reverse Engineering Assessments and Methods 2 (DREAM2) contest, we sought to reconstruct the five-gene network for which qPCR time-series data were provided in the form of two independent experiments. To impose consistency in the reconstructed model from the two time series, we chose to regress the two experiments simultaneously. To our disappointment, we found high residuals in the fits of the temporal trajectories, leaving ~50% of the variance unexplained, even in the normally overly accurate OLS methods. Moreover, it could be that the near–steady state prerequisite and linearity of the model are invalid approximations as pointed out for the repressilator in oscillator regime. Nonetheless, we think that important features of the network have been captured and shared by the four algorithms. The most evident is the strong induction of gene A (node 1) by gene C (node 3), followed by the parallel

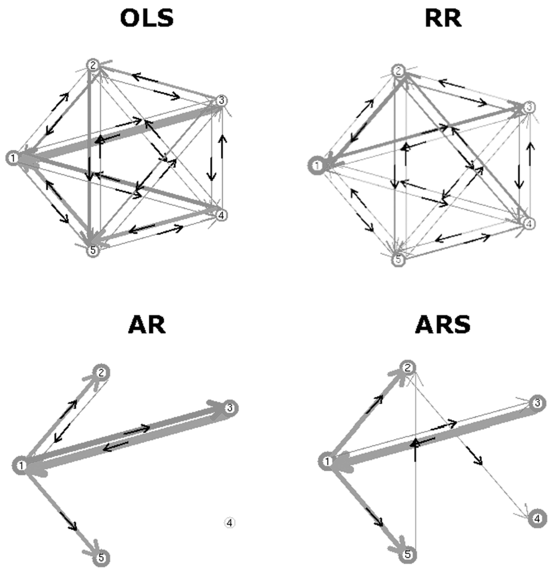

FIGURE 4. Reconstruction by the four algorithms of the DREAM2 five-gene network from qPCR measurement. The common features are induction by gene A (node 1) on its neighbors E and B (nodes 5 and 2) and the induction of gene A (node 1) by gene C (node 3) with a negative feedback from A (node 1) to C (node 3).

induction of genes B (node 2) and E (node 5) by gene A. Repression of gene C by gene A, likely to act as a feedback mechanism, also has a strong signature in all the inferred networks (Fig. 4).

Discussion

Biochemical networks constitute a formidable algorithmic challenge for many reconstruction methods. In this paper we compared and extended previous penalized regression algorithms for RR and for AR for the reconstruction of genetic networks from time-series data. Despite many caveats already discussed, linear regression models have the advantage of being computationally tractable: OLS and RR optimization problems (even constrained) are well posed and convex, so that global optima can be found in reasonable time. In contrast, the global convergence of AR and ARS is not always guaranteed.[22] It may be that this is one of the reasons for the long tails in the relative residual variance of the AR reconstructed networks, and several initial conditions should be tested. Though convergence to a global minimum may not be always guaranteed, we have shown the benefits of AR in the reconstruction of biological networks. In contrast to OLS and RR, AR naturally provides sparse solutions without the need for *a posteriori* thresholds: the balancing of costs for each parameter allows one to strongly penalize some parameters while releasing the selective pressure on the more significant ones. Moreover the specific implementation using the hyperparameter λ^{22} to tune the sparseness of the inferred Jacobian makes it suitable for cross-validation methods. One aspect of the methodology that could potentially lead more accurate reconstructions concerns the sparseness condition in the continuous to discrete time representations of dynamical systems as discussed in the Material and Methods section.

Imposing biological constraints is often invoked in the context of data modeling and integration. We proposed to enforce the sign of the outgoing regulatory links to be consistent for a given gene. We thus proposed an extension of AR, termed ARS, which implements the sign coherence in the regulatory activity of each node. For small networks, we can enumerate and scan all possible combinations of constraints, while minimizing the same regression functions. In the benchmark, ARS has proven to be more accurate than the other three methods in identifying the correct links. Unfortunately, the current implementation is affected by a long execution time, suggesting further development of iterative heuristic methods to bypass the brute force scanning of all possible sign combinations.

The four algorithms have been applied to the reconstruction of dynamical systems from *in silico*–generated data. The complexity of interactions achievable with the differential equations describing these systems is beyond the capabilities of linear models, yet an interesting number of features have been correctly identified. In the four-gene network presented by Ref. 16, AR inferred correctly 75% of the links, missing two out of the four activating interactions. Interestingly, wrong direct feedback repressions (from node 1 and 2 to node 4) were added to the two activating links, maybe a mechanism by which our proposed linear model reinterpreted saturation of node 4. However, in the case of the repressilator, none of the three algorithms, OLS, RR, and AR, managed to infer more than 40% of the real network correctly. ARS, the extension of AR implementing sign constraints, inferred a sparse network, correctly identifying mRNA translations. Yet repression acting via saturation was not captured, and extra links were proposed by all methods to cope with the high nonlinearity of the underlying model. Further extensions, such as the implementation of sigmoidal transfer functions, will be needed to overcome these current limitations, but the associated nonlinear optimization problems will be highly challenging. Overall, ARS proved outstanding when compared to OLS and RR in inferring the topology of randomly generated dynamical systems, making it a promising tool to aid the network reconstruction and interpretation of time-series measurements.

Material and Methods

Model

The N dimensional dynamical state $X(t) = (X_1(t), ..., X_N(t))$ is assumed to obey the following linear dynamical system:

$$\begin{aligned} \tfrac{dX}{dt} &= MX + B \\ Y_t &= X_t + \eta_t \end{aligned} \qquad (1)$$

where the second equation represents the effect of the measurement in terms of equidistant sampling and acquisition error. The noise is taken to be uncorrelated Gaussian white noise with variance σ^2. Intrinsic cellular noise averages out in population measurement such as those from qPCR, which is what we are interested in. In other words it is well justified to neglect dynamical noise propagation. The linear model is well suited for perturbations off a steady state (see discussion about extensions of the model). As for every continuous-time linear dynamics an equivalent discrete-time model can be found, we can assume the following model for our sampled data

$$\begin{aligned} X_{t+1} &= \tilde{M}X_t + \tilde{B} \\ Y_t &= X_t + \eta_t \end{aligned} \qquad (2)$$

where $\tilde{M}$ and $\tilde{B}$ need to be estimated. To solve for the unknowns we switch to vector notation using the Kronecker and *vec* operations:

$$\begin{aligned} Z &= vec(Y) \\ A &= \left(X^T \otimes I_N, 1_T \otimes I_N\right) \\ \beta &= \left(vec(\tilde{M}), \tilde{B}\right) \end{aligned} \qquad (3)$$

which allows to write the system as

$$Z = A\beta + \varepsilon, \text{ where } Z = Z(Y), A = A(Y), \beta = \beta(\tilde{M}, \tilde{B}).$$

Dimension of Y is $N \times T$, and A is $(N \times T) \times (N^2 + N)$, where T is the number of time points in the series. The noise in Y affects both the data Z and the design matrix A. Thus, in theory, TLS regression would seem appropriate; however, an important result states that when total least square is combined with penalized regression, the estimation is equivalent to the ordinarily penalized regression.[20] Therefore we will proceed as if the noise η_t acts on Z and not on A.

Penalized Regressions

The following regressions were performed. The objective functions S are given for each case.

- (OLS) Ordinary least squares $S(\beta) = ||Z - A\beta||^2$
- (RR) Ridge regression $S(\beta) = ||Z - A\beta||^2 + \lambda||\beta||^2$
- (AR) Adaptive ridge regression $S(\beta) = ||Z - A\beta||^2 + \sum_i \lambda_i\beta_i^2, \frac{1}{N^2}\sum_i \frac{1}{\lambda_i} = \frac{1}{\lambda},$

where N is the number of columns in the Jacobian, λ is a hyperparameter tuning the global sparseness, and the λ_i's are tuned adaptively following the expectation-maximization algorithm proposed in Ref. 22. In this scheme the parameters are updated iteratively: at each step s the optimal parameters λ_i^s of the Bayes prior are estimated from $\beta_i^{(s-1)}$ and then the posterior is maximized to compute $\beta_i^{(s)}$. Coefficients β_i's whose associated λ_i diverges during this procedure are pruned leading to sparse solutions. The method was shown to lead to solutions that are equivalent to the L1 penalized regression or Lasso.[21] Seeding is done using OLS, as suggested in Ref. 22.

- (ARS) Adaptive ridge with sign constraints regression implements the same minimization as in the AR, but further imposes the constraint that nondiagonal elements have the same sign in any given column: $S(\beta) = ||Z - A\beta||^2 + \sum_i \lambda_i \beta_i^2$, $\frac{1}{N^2}\sum_i \frac{1}{\lambda_i} = \frac{1}{\lambda}$, *subject to* $\tau_j \beta_i > 0, j = 1...N,\ i \neq 1 + (j-1)(N+1),\ \tau_j \in \{-1;1\}$ Here we implement an exhaustive approach: we use quadratic programming to compute the regression for each 2^N sign combination. The retained ARS solution is the one showing the smallest cross-validation error. However, it is possible to extend the formulation of the sign constraints to allow unsigned columns: in this case $\tau_j \in \{-1;0;1\}$; the 3^N sign combinations of this latter formulation make the exhaustive search computationally intensive for larger systems.

Sparseness and the Correspondence between Discrete and Continuous Dynamics

With the above method of penalized regression, we favor coefficient vectors β_i with small norm. In particular, for the case of AR and ARS, we impose sparseness on the solution. As the solution vector corresponds to the discrete-time gene connectivity matrix $\tilde{M}$, this sparsity constraint appears to be plausible. But what is the relation between the sparseness of $\tilde{M}$ and the sparseness of the continuous-time matrix M? To clarify this one has to develop the correspondence between a continuous-time linear system and its discrete-time counterpart. For the system (1) we can solve for the time evolution as

$$X(t+\Delta t) = \exp(M\Delta t)$$
$$\exp(Mt)X(0) - M^{-1}B,$$

which allows us to define the recursion

$$X(t+\Delta t) = \exp(M\Delta t)X(t)$$
$$+(\exp(M\Delta t) - 1)M^{-1}B,$$

such that we find the exact correspondence $\tilde{M} = \exp(M\Delta t)$ and $\tilde{B} = (\exp(M\Delta t) - 1)$ $M^{-1}B$. Thus, sparseness on $\tilde{M}$ does not necessarily lead to sparseness in $M = \log(\tilde{M})/\Delta t$. Nevertheless, the rationale of our above approach to penalize the norm of $\tilde{M}$ is twofold. First, with penalizing the norm of the continuous-time matrix M, one would leave the realm of linear regression, as the penalty term in the cost function is not a quadratic function in the coefficient vector β_i anymore. Thus, for the resulting optimization problem issues such as nonuniqueness of the solution and local minima would come into play. Second, linearization of the correspondence between the continuous and discrete world by performing a first-order approximation of the above matrix logarithm gives

$$M \approx M^* = (\tilde{M} - I)/\Delta t. \qquad (4)$$

The relation exactly represents the forward Euler method, a frequently applied method for numerical integration of differential equations leading to a discrete system of the form

$$X_{t+1} = (I + \Delta t M^*)X_t + \tilde{B}$$
$$Y_t = X_t + \eta_t.$$

Evidently from (4), one can see that imposing sparseness on $\tilde{M}$ also implies sparseness of the continuous-time system matrix M. The

Euler method gives a good approximation if the time constants of the continuous dynamics are large compared to the sampling time of data acquisition.

Cross Validation

For the RR, AR, and ARS methods, the optimal hyperparameter λ is determined with v-fold cross validation.

The full set of time points is split into training and testing sets such that the fraction of points in the test set equals v/T. The training set is used to infer the model (M^*,B^*).

Given that time points appear both in $\mathcal{Z}$ and A due to the regression structure (3), we retain all instances of a time point from $\mathcal{Z}$ and A. The test error is calculated as the total residual variance in the time trajectories. Resampling of the training and test sets is repeated multiple times, and the median of all testing error is defined as the cross-validation error.

The optimal penalizing hyperparameter is the one associated with the smallest cross-validation error.

Throughout the manuscript we used a 5-fold cross validation with 500 resamplings. Once the optimal cross-validation parameter λ is found we retrain the model on the full set of time points.

Assessing the Accuracy of the Reconstruction

We consider two error measures for the accuracy of the Jacobian M^*.

The first is the relative Frobenius norm $F_{M^*} = \|M - M^*\|^2 / \|M\|^2$ where M is the real and M^* the reconstructed matrix. This measure represents the relative amount of variance of the matrix M captured by the inferred Jacobian.

The second metric, termed binary norm, reads $\varepsilon_{M^*} = \sum_{ij} \delta_{ij}$, with

$$\delta_{ij} = \begin{cases} 0 & \mathit{if}\ \mathit{sign}(M_{ij}) = \mathit{sign}(M^*_{ij}) \\ 1 & \mathit{if}\ \mathit{sign}(M_{ij}) \neq \mathit{sign}(M^*_{ij}) \end{cases}$$

This metric reflects strictly the topology of the network rather than magnitudes of links.

Random Networks

Random matrices and vectors are generated as follows. The N dimensional intercept vector $\tilde{B}$ is sampled from a uniform distribution [0,1]. The matrix $\tilde{M}$ is designed to have negative elements along the diagonal and a predefined number of nonzero off-diagonal elements drawn from a uniform [0,1] distribution, such as seven elements for the networks in Figures 1 and 2. The sign along each column is fixed, excluding the diagonal. This reflects the biological requirement that, in the majority of cases, transcription regulators in lower organisms act consistently as either activators or repressors, but not both.

We further impose some regularity constraints on the stability of the model: $\tilde{M}$ is considered suitable if the eigenvalues of $\tilde{M}$ lie within the complex unit circle. For each suitable pair ($\tilde{M}$,$\tilde{B}$), trajectories are generated according to the recursion rule (2). To reflect the experimental situation, 20 time steps are used, and the spacing is chosen such that the slowest decay mode reaches 0.1% of its steady states after the 20 time points. This ensures that we cover the transient parts of the trajectories that contain the information about the dynamics.

Gaussian white noise σ^2 is added to the trajectories before reconstructing the model. The noise variance σ^2 is set to be a fraction k of the total variance var(X_t), typically we use 5 or 10%.

Four-Gene Network by Sontag and Repressilator Models

The four-gene network reconstructed in Figure 3A was taken from Ref. 16 using the original parameters: $V^s_1 = 1$, $A_{14} = 4$, $K^a_{14} = 1.6$, $n_{14} = 2$, $K^I_{12} = 0.5$, $n_{12} = 1$, $V^s_2 = 0.7$, $A_{24} = 4$, $K^a_{24} = 1.6$, $n_{24} = 2$, $V^s_3 = 0.6$, $A_{32} = 5$, $K^a_{32} = 1.5$, $n_{32} = 2$, $K^I_{31} = 0.7$, $n_{31} = 1$, $V^s_4 = 0.8$, $A_{43} = 2$, $K^a_{43} = 0.15$, $n_{43} = 2$,

$V_1^d = 40$, $K_1^d = 30$, $V_2^d = 100$, $K_2^d = 60$, $V_3^d = 30$, $K_3^d = 10$, $V_4^d = 100$, $K_4^d = 50$. Initial conditions were set to zero for each gene.

Trajectories were simulated in the interval $t \in [0, 20]$ and discretized to provide 20 equally spaced time points.

The repressilator model was taken from Ref. 24 with asymmetric parameters $n = 2$, $b_j = \{5;4;3\}$, $a_j = \{8;5;4\}$, $a_{0j} = \{0.1;0.5;1\}$ and initial conditions corresponding to 10 proteins of each gene and no mRNA. These parameter choices generate damped oscillations converging to a stable node. The trajectories contained 20 time points spanning two oscillations.

DREAM2 Data

We merged the two time-course qPCR measurements provided by the DREAM2 stacking the respective vectors Z and the design matrices A columnwise. The regression structure (3) does not require time continuity between adjacent elements in the vector Z, thus merging is possible without creating artifacts.

The data were fit in natural units, that is, absolute expression values where used as the X(t) variable.

Conflicts of Interest

The authors declare no conflicts of interest.

References

1. Gasch, A.P. *et al.* 2000. Genomic expression programs in the response of yeast cell to environmental changes. *Mol. Bio. Cell.* **11:** 4241–4257.
2. Crampin, E.J., S. Schnell & P.E. McSharry. 2004. Mathematical and computational techniques to deduce complex biochemical reaction mechanisms. *Prog. Biophys. Mol. Biol.* **86:** 77–112.
3. Eisen, M.B. *et al.* 1998. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA* **95:** 14863–14868.
4. Wagner, A. 2001. How to reconstruct a large genetic network from n gene perturbations in fewer than n^2 easy steps. *Bioinformatics* **17:** 1183–1197.
5. Werhli, A.V., M. Grzegorczyk & D. Husmeier. 2006. Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical Gaussian models and Bayesian networks. *Bioinformatics* **22:** 2523–2531.
6. Arkin, A., P. Shen & J. Ross. 1997. A test case of correlation metric construction of a reaction pathway from measurements. *Science* **77:** 1275–1279.
7. Samoilov, M., A. Arkin & J. Ross. 2001. On the deduction of chemical reaction pathways from measurements of time series of concentrations. *CHAOS* **11:** 108–114.
8. Margolin, A.A. *et al.* 2006. ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics* **7:** S1–S7.
9. Friedman, N. *et al.* 2000. Using Bayesian networks to analyze expression data. *J. Comput. Biol.* **7:** 610–620.
10. Rangel, C. *et al.* 2001. Modelling biological responses using gene expression profiling and linear dynamical systems. *Proc. Pacific Symposium on Biocomputing* **2001:** 248–256.
11. Laubenbacher, R. & B. Stigler. 2004. A computational algebra approach to the reverse engineering of gene regulatory networks. *J. Theor. Biol.* **229:** 523–537.
12. Shmulevich, I. *et al.* 2002. Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics* **18:** 261–274.
13. Weaver, D.C., C.T. Workman & G.D. Stormo. 1999. Modeling regulatory networks with weight matrices. *Pacific Symposium on Biocomputing* **4:** 112–123.
14. D'haeseleer, P. *et al.* 1999. Linear modeling of mRNA expression levels during CNS development and injury. *Proc. Pacific Symposium on Biocomputing* 41–52.
15. Liao, J.C. *et al.* 2003. Network component analysis: Reconstruction of regulatory signals in biological systems. *Proc. Natl. Acad. Sci.* **100:** 15522–15527.
16. Sontag, E., A. Kiyatkin & B.N. Kholodenko. 2004. Inferring dynamic architectures of cellular networks using time series of gene expression, protein and metabolite data. *Bioinformatics* **20:** 1877–1886.
17. Kim, J. *et al.* 2007. Least-squares methods for identifying biochemical regulatory networks from noisy measurements. *BMC Bioinformatics* **8:** 8.
18. Rogers, S. & M. Girolami. 2005. A Bayesian regression approach to the inference of regulatory networks from gene expression data. *Bioinformatics* **21:** 3131–3137.
19. Tegner, J. *et al.* 2003. Reverse engineering gene networks: Integrating genetic perturbations with dynamical modelling. *Proc. Natl. Acad. Sci.* **100:** 5944–5949.
20. Golub, G.H., P.C. Hansen & D.P. O'Leary. 1999. Tikhonov regularization and total least squares. *SIAM J. Matrix Analysis Appl.* **21:** 185–194.

21. Tibshirani, R.J. 1995. Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc.* **B 58:** 267–288.
22. Grandvalet, Y. & S. Canu. 1999. Outcomes of the equivalence of adaptive ridge with least absolute shrinkage. *Proceedings of the 1998 Conference on Advances in Neural Information Processing Systems II*, pp. 445–451.
23. Salgado, H. *et al*. 2006. RegulonDB (version 5.0): Escherichia coli K-12 transcriptional regulatory network, operon organization, and growth conditions. *Nucleic Acids Res.* **34:** D394–D397.
24. Elowitz, M.B. & S. Leibler. 2000. A synthetic oscillatory network of transcriptional regulators. *Nature* **403:** 335–338.